IN THE CLAIMS:

Please amend claims 1, 14, 27 and 28, as indicated hereinbelow.

Claims 2-13, claims 15-26 and claims 29-40 are dependent claims which remain "*as is*".

1. (Currently Amended) A method for adapting a legacy software application, created from legacy source code and developed for an environment comprising a centralized computing resource interconnected to a series of computer terminal devices, to a network environment, wherein said network environment comprises a system of distributed, interconnected network computing resources, said method comprising the steps of:

~~utilising~~ providing a software application which utilizes said legacy source code to automatically produce a series of executable software components that provide the functionality for interaction with [[the]] said legacy software application, said components being executable by at least one of said computing resources in said network environment, and wherein upon execution, said computing resource is caused to interconnect with said legacy software application over said network so as to interact with said legacy software application in the transmission or receipt of information to and from said legacy software application.

2. (Original). A method in accordance with claim 1 wherein the legacy software application includes interface specification definitions which include definitions of screen formats, the step of producing the series of software components further comprising generating a series of user interface software components from the screen format definitions, the user interface software components being arranged for execution on the network computing resource to provide a graphical user interface providing at least data entry and display facilities of the interface specification definitions.

3. (Original). A method in accordance with claim 2, wherein the interface software components are arranged to generate forms corresponding to forms generated by the legacy software application.

4. (Previously Presented). A method in accordance with claim 1, comprising the step of generating client interface components, the client interface components being arranged to interact over the network with the legacy software application.

5. (Original). A method in accordance with claim 4, the client interface components include a user input object which is arranged to receive data input by a user of the network computing resource and transmit the data to the legacy application, over the network.

6. (Previously Presented). A method as claimed in claim 1, wherein said series of software components are loadable and executable by an Internet Browser.

7. (Previously Presented). A method as claimed in claim 1, any previous claim wherein said series of software components comprise Java code applets.

8. (Previously Presented). A method as claimed in claim 1 wherein said series of software components are executable by scripting languages running on said network computing resource.

9. (Previously Presented). A method as claimed in claim 1, wherein said source code includes a series of data fields and said series of software components include object oriented methods for setting or obtaining values of said series of data fields.

10. (Previously Presented). A method as claimed in claim 1, wherein said network environment comprises the Internet network.

11. (Previously Presented). A method as claimed in claim 1, wherein said network environment utilizes TCP/IP transfer protocols.

12. (Previously Presented). A method as claimed in claim 1, wherein said source code is written in a 4GL language.

13. (Original). A method as claimed in claim 12 wherein said source code is written in the LINC language.

14. (Currently Amended) A program storage device readable by a machine, tangibly embodying a program of instructions executable by [[the]] said machine to perform method steps for adapting a legacy software application created from legacy source code and developed for an environment comprising a centralized computing resource interconnected to a series of computer terminal devices, to a network environment, wherein said network environment comprises a system of distributed, interconnected network computing resources, said method comprising the steps of:

utilizing providing a legacy software application which utilizes said legacy source code to automatically produce a series of executable software components that provide the functionality for interaction with [[the]] said legacy software application, said components being executable by at least one of said computing resources in said network environment, and wherein upon execution, said computing resource is caused to interconnect with said legacy software application over said network so as to interact with said legacy software application in the transmission [[of]] or receipt of information to and from said legacy software application.

15. (Original). A program storage device in accordance with Claim 14, wherein the legacy software application includes interface specification definitions which include definitions of screen formats, the method further comprising the step of generating a series of user interface software components from the screen format definitions, the user interface software components being arranged for execution on the network computing resource to provide a graphical user interface providing at least data entry and display facilities of the screen format.

16. (Original). A program storage device in accordance with claim 15, wherein the interface software components are arranged to generate forms corresponding to forms generated by the legacy software application.

17. (Previously Presented). A program storage device in accordance with claim 14, the method further comprising the step of generating client interface components, the client interface components being arranged to interact over the network with the legacy software application.

18. (Original). A program storage device in accordance with claim 17, wherein the client interface components include a user input object which is arranged to receive data input by a user of the network computing resource and transmit the data to the legacy application, over the network.

19. (Previously Presented). A program storage device in accordance with claim 14, wherein said series of software components are loadable and executable by an Internet Browser.

20. (Previously Presented). A program storage device in accordance with claim 14, wherein said series of software components comprise Java code applets.

21. (Previously Presented). A program storage device in accordance with claim 14, wherein said series of software components are executable by scripting languages running on said network computing resource.

22. (Previously Presented). A program storage device in accordance with claim 14, wherein said source code includes a series of data fields and said series of software components include object-oriented methods for setting or obtaining values of said series of data fields.

23. (Previously Presented). A program storage device in accordance with claims 14, wherein said network environment comprises the Internet network.

24. (Previously Presented). A program storage device in accordance with claim 14, wherein said network environment utilized TCP/IP transport protocols.

25. (Previously Presented). A program storage device in accordance with claim 14, wherein said source code is written in the LINC language.

26. (Previously Presented). A program storage device in accordance with claim 14, wherein said terminal screen definitions are written in a screen control language.

27. (Currently Amended) A method for adapting a 4GL legacy software application including template definitions from which a legacy software application can be generated, comprising the steps of:

~~utilizing~~ providing a software application which utilizes said template definitions to automatically produce a series of software executable components, said components being executable by at least a computing resource in a network environment comprising a system of distributed, interconnected network computing resources, and wherein upon execution, said computing resource is caused to interconnect with [[the]] said 4GL legacy software applications so as to interact with [[the]] said legacy software application in the transmission and receipt of information to and from [[the]] said legacy application.

28. (Currently Amended) A system for adapting a legacy software application, created from legacy source code and developed for an environment comprising a centralized computing resource interconnected to a series of computer terminal devices, to a network environment, wherein said network environment comprises a system of distributed, interconnected network computing resources, the system comprising;

means utilizing said legacy source code to automatically produce a series of executable software components for providing the functionality for interaction with [[the]] said legacy software application, said components being executable by at least one of said computing resources in said network environment, and wherein upon execution, said computing resource is caused to interconnect with said legacy software application over said network so as to interact with said legacy software application in the transmission or receipt of information to and from said legacy software application.

29. (Original). A system as claimed in claim 28, wherein the legacy software application includes interface specification definitions which include definitions of screen formats, the means for producing this series of software components including means for generating a series of user interface software components from the screen format definitions, the user interface software components being arranged for execution on the network computing resource to provide a graphical user interface providing at least data entry and display facilities of the interface specification definition.

30. (Original). A system in accordance with claim 29, wherein the interface software components are arranged to generate forms corresponding to forms generated by the legacy software application.

31. (Previously Presented). A system in accordance with claim 28, the means for producing the series of software components including means for generating client interface components, the client interface components being arranged to interact over the network with the legacy software application

32 (Previously Presented). A system in accordance with claim 31, the client interface components including a user input object which is arranged to receive data input by a user of the network computing resource and transmit the data to the legacy application, over the network.

33. (Previously Presented). A system in accordance with claim 28, wherein said series of software components are loadable and executable by an Internet Browser.

34. (Previously Presented). A system in accordance with claim 28, wherein the series of software components comprise Java code applets.

35. (Previously Presented). A system in accordance with claim 28, wherein said series of software components are executable by scripting languages running on said network computing resource.

36. (Previously Presented). A system in accordance with claim 28, wherein said source code includes a series of data fields and said series of software components include object orientated methods for setting or obtaining values of said series of data fields.

37. (Previously Presented). A system in accordance with claim 28, wherein said network environment comprises the Internet network.

38. (Previously Presented).   A system in accordance with claim
28, wherein said network environment utilizes TCP/IP transfer
protocols.

39. (Previously Presented).   A system in accordance with claim
28, wherein said source code is written in at 4GL language.

40. (Previously Presented).   A system in accordance with claim
28, wherein said source code is written in the LINC language.